
研究報告

裁判例データベースを用いた新たな研究手法の開発

法学部法学科 教授 山下 純 司
計算機センター 教授 久保山 哲 二

1 研究の目的等

(1) はじめに

わが国には、複数の裁判例のデータベースが存在し判決文も電子データとして公表されているものが多数存在する。しかし、そうした裁判例を大量に使ったテキストマイニングなどの研究手法は、現在のところほとんど行われていない。その原因には、さまざまなものが考えられるが、第一に、データベースが人による操作のみを想定して作られていること、第二に、判決文自体が定型性がなく画一的な処理に不向きであること、第三に、研究分野として未開拓のためその実用性等が検証されていないことなどが挙げられる。

本研究では、裁判例データベースから、判決文のデータを取得し、そのデータを機械学習で処理しやすいよう、また人間がより読みやすくなるような工夫をすることによって、将来的にテキストマイニング等の手法による新たな法学研究の可能性を探求することを目指している。

2016 度は裁判例データベースからの判決文データの効率的な取得を中心に研究を行い、一定の成果をあげた。そこで、2017 年度は、得られた判決文データを機械学習で読み取らせることで、内容を自動判別する工夫について、研究を行った。

(2) 判決文の構造

わが国の裁判所によって下される判決文は、一定の様式に従って書かれている。判決文は、国家機関としての裁判所の命令である主文と呼ばれる部分と、その命令がどのような根拠に基づき下されたのかを明らかにする理由と呼ばれる部分に分かれている。主文は、判決の中心部分であり、裁判の元になった紛争に対して国家がどのような解決をするのかを明らかにするものであるから、もともと重要な部分であるが、その文章は簡潔なものであり、結論しか書かれていないので、どのような紛争があつて判決が下されたのかは主文を読んでもわからない¹。そうした点は、判決文の理由中に記載される。

判決の理由は、最高裁判所と下級審裁判所とは、やや内容が異なる。下級審裁判所では、その紛争についての複数の争点について、原告の主張、被告の反論が紹介され、最終的に裁判所

¹ 例えば「被告は、原告に対し、金〇〇円を支払え。」といった文章や、「原告の請求を棄却する。」といった文章である。

がそれらの争点について、どう判断したのかが説明されている。事件が複雑になるほど、争点が増えて、理由も長くなるのが普通である。こうした判決理由を正確に読むのは、実は訓練を受けた法律家でも難しい。原告、被告の主張が交互に出てくるうえ、裁判所の説明も厳密さを求めるが故に回りくどい。法律に詳しくない一般人が判決理由を正確に理解することは極めて難しい。

(3) 判決文を読みやすくする工夫

そこで、判決理由を読みやすい形に整形することができれば、法律の初学者だけでなく、専門家にとっても、学習や業務の効率化に役立つであろう。例えば、判決理由中の原告の主張、被告の主張、裁判所の判断を区別して、自動的に色分けをしてくれるだけでも、判決理由が読みやすくなると思われる。

機械学習によって、原告の主張、被告の主張、裁判所の判断を区別する技術は、判決文をテキストマイニングの技術によって分析するという本研究の将来的な目標にも役立つ。法学の研究にとって、最も重要なのは裁判所の判断なので、その部分だけを抽出して分析をすることができれば、分析の精度が上がることを期待できるからである。

(4) 利用した判決文のデータ

本研究で利用したのは、昨年度(2016年度)の研究で取得した約500件の判決文である。これらの判決文は、独自に開発したクローラーのプログラムを使ってデータベースから大量入手した。この判決文を、法学部の学生アルバイトに依頼して、「原告の主張」、「被告の主張」、「裁判所の判断」など、内容ごとに色分けを行い、さらに色分けデータを html 形式のタグ付けデータに変換した(変換プログラムは、外部委託で作成した)。

表 1. 判決文のデータ数

	訓練データ	テストデータ	合計
判決文	400	91	491
判決文に含まれる文の総数	95338	19290	114628

(5) 機械学習としての問題設定と実験結果

前節(4)で示したように、人手によって判決文に含まれる文章を、役割に応じて「原告の主張」、「被告の主張」、「裁判所の判断」、「その他」に分類し、これを教師データとして深層学習による文章のラベル付けを行った。実験では、あらかじめデータを訓練データとテストデータに分けて性能評価を行った。表 1 に実験に用いた判決文に含まれる文章の数を示す。また、図 1 に人手によ

て文章にラベル付けした分類項目の内訳を示す。

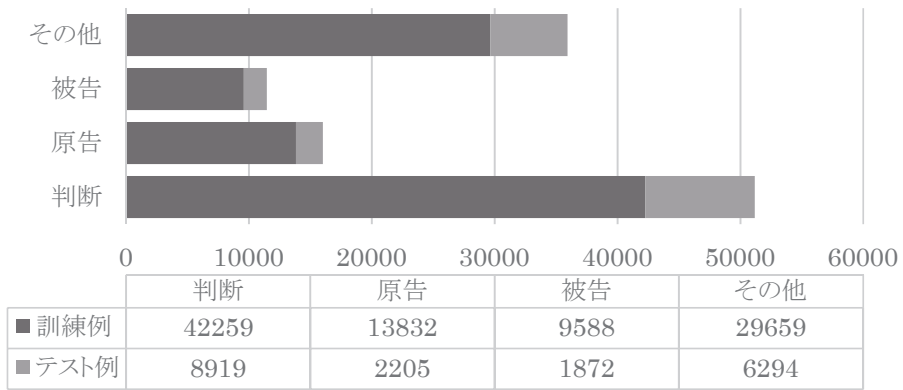


図 1. 分類ラベルの分布

訓練データによる機械学習には、リカレントニューラルネットワーク Bidirectional GRU(Gated Recurrent Unit)[1]によるモデルを用いた(図 2 参照)。図において、文1, 文2は、入力データにおいて、隣接する2つの文であり、 w_1, \dots, w_{t-1} は、各々の文における形態素の並びを表す。なお、形態素は、形態素解析システム JUMAN++[2]を用いて分割した。また、図中で記号 h_i は隠れ層を表している。各々の文をニューラルネットワークによって抽象化した後、前後の文脈を考慮して、各文に付けられるラベルを出力する。実行環境は Intel Core i5-7500T(2.7GHz, Quad Core), 16GB RAM, NVIDIA GeForce GTX1070 の Linux マシンであり、PyTorch によりモデルを実装した。訓練データにより訓練したモデルにより、テストデータを評価したところ、4 分類で約 85% の精度が得られた。

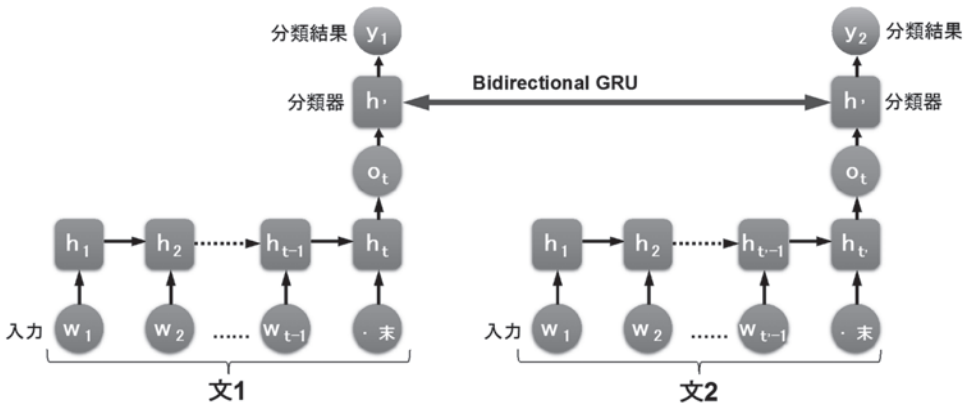


図2. 学習に用いたモデル

(6) 今後の課題

今回の実験では当初想定していたよりも高い精度が得られた。しかしながら、いくつかの課題がある。まず、実用に供するには 85%程度の精度ではまだ十分でない。また、精度が想定以上に高かった理由として、実験データと訓練データの分割を完全にランダムにおこなったため、同じ裁判官の執筆した判決文が、訓練データとテストデータにまたがって存在していることが挙げられる。そのため、既存の判決文が存在していない裁判官の判決文との比較をおこない検証する必要がある。また、依然として訓練データが不足しており、今後継続して人手により分類ラベルを付したデータを増やしてゆく必要がある。

参考文献

- [1] Dzmitry Bahdanau, Kyunghyun Cho, Yoshua Bengio. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.
- [2] Hajime Morita, Daisuke Kawahara and Sadao Kurohashi: Morphological Analysis for Unsegmented Languages using Recurrent Neural Network Language Model, Proceedings of EMNLP 2015: Conference on Empirical Methods in Natural Language Processing, pp.2292-2297, 2015.